



Generando proyecto

Transcripción

[00:00] Bien. Vamos a comenzar. Entonces, para crear nuestro proyecto en Spring le vamos a hacer mediante la herramienta que el mismo Spring nos está proporcionando, el cual es el Spring Initializr. Spring Initializr ustedes lo pueden poner en cualquier buscador, y en la primera opción `start.spring.io`, entramos aquí y vemos que es básicamente un formulario, un formulario que Spring nos da para elegir qué dependencias, qué características queremos que tenga nuestro proyecto.

[00:28] Por ejemplo la parte de proyecto vemos que nos dice quieren que sea Gradle con lenguaje Groovy, Gradle con lenguaje Kotlin o Maven para lo que es gestión de dependencias, como ya les comenté en el video anterior, vamos a coger Maven y lenguaje, obviamente va a ser Java.

[00:46] Para la versión de Spring Boot vamos a usar la 3.0.2, no necesitamos discutir sobre las versiones aquí. La versión 2.7 es un poco antiguo para lo que se usa actualmente, así que vamos a partir por la que está seleccionada por defecto en momento, inicio de 2023, es la 3.0.2.

[01:04] En la metadata del proyecto es donde tenemos que comenzar a poner ya las características que tenga nuestro proyecto como nombre de paquetes, el nombre del artefacto, entre esas cosas. Por ejemplo, en el grupo le vamos a poner `med.voll` porque va a ser la clínica que vamos a usar. En el artefacto vamos a ponerle `api`. El nombre también va a ser `api`.

[01:30] En la descripción de proyecto vamos a poner API rest para Voll clínica.Y el nombre del paquete va a ser med.voll.api. Automáticamente lo generó, pero si yo quiero puedo cambiarlo, editarlo a como yo más lo prefiera. Por buena práctica, lo voy a mantener así, porque es como se debería llamar, entonces lo dejamos tal cual.

[02:00] Y aquí en packaging la forma en que vamos a empaquetar nuestro proyecto nos dice un jar o un war. Antiguamente, teníamos lo que se llama servidores de aplicaciones. Ahí lo que tú hacías era coger el archivo .war, desplegarlo en el servidor de aplicaciones y solo en ese momento tu proyecto era accesible.

[02:19] Pero ya que estamos tratando con Spring Boot, Spring Boot ya nos da la opción de un servidor dentro de la misma aplicación, por lo tanto no necesitamos que sea un archivo .war sino tranquilamente ejecutar un jar. Esto ya lo pueden saber, pero solamente para reforzar un poco la idea de Spring Boot.

[02:37] Sobre Java la versión 17 es la más usada actualmente. Hay muchas empresas que están migrando de la 11 a la 17 por el soporte que está dando, es un tipo de soporte más largo de parte de Oracle, entonces vamos a quedarnos con la versión 17. La 19 es buena, es estable, pero no es la más usada en el mercado actualmente, como les repito, enero 2023. Entonces nos quedamos con la 17. No se preocupen.

[03:02] La parte derecha del formulario ya contiene lo que son las dependencias. Esto quiere decir qué cosas va a usar nuestro proyecto externas a lo que tenemos aquí. Si le ponemos aquí en agregar dependencias, vamos a ver un montón de opciones, que podemos agregarle a nuestro proyecto, por ejemplo, vemos que tenemos soporte para GraalVM, no vamos a usarlo. Spring Boot DevTools, esto sí lo vamos a usar porque nos va a dar la facilidad de modificar nuestro código y no tener que reiniciar el servidor.

[03:33] Para que veamos los cambios en tiempo real. Entonces le damos que sí. Vemos que ya lo va agregando Spring Boot DevTools, nuevamente agregar dependencias. ¿Qué más deberemos agregar? Lombok, como les mencioné es una herramienta que la van a amar. Entonces sí, claro que sí, vamos con Lombok. ¿Y después que más le vamos a agregar?

[03:51] Vamos por aquí Spring Web, que si bien no vamos hacer una aplicación web, ya lo explicamos en el video introducción, pero Spring Web nos da las librerías que necesitamos para exponer nuestros métodos de API Rest a través de post, get, put, delete, etcétera. Entonces vamos a usar en efecto Spring Web, seleccionamos.

[04:11] Y ya tenemos tres dependencias, vamos a explorar las otras, por ejemplo, tenemos Spring Reactive, Spring for GraphQL y otras herramientas que por el momento me parece que no van a ser necesarias, Jersey no lo vamos a usar, porque ya tenemos Spring Web, entonces no se preocupen, no necesitamos Jersey.

[04:31] Spring Security, vemos que tenemos una gran cantidad de opciones propias de Spring y librerías de terceros en algunos casos para poder generar nuestro proyecto. No vamos a marearnos más con cada una de estas opciones. Vamos a cerrar eso por el momento y finalmente generar nuestro proyecto.

[04:55] Para eso podemos darle clic a generar o apretar “Command + Enter” si estás en una Mac o el atajo que te indique aquí si estás en una computadora Windows o Linux. En este caso yo voy a dar clic, damos a generar y lo que va a ser eso es descargar nuestro proyecto en nuestra computadora.

[05:13] Como lo está descargando, lo que tengo que hacer ahora es acceder a finder y vemos que aquí está mi api.zip. Este es el proyecto que acaba de descargar de Spring Initializr. Lo que voy a hacer ahora es descomprimir. Entonces vamos a abrir con, esta es mi herramienta para descomprimir y vemos que aquí está ya API descomprimido.

[05:41] Nuevamente exploren las opciones que tenemos aquí, si es que le resulta interesante. No olviden que la mejor forma de aprender es practicando. Entonces no se preocupen, no tengan miedo. Spring va a crearles un proyecto bien configurado, por el momento esto es lo que necesitamos, ya tenemos el proyecto. Ya lo tenemos en nuestra computadora. En el siguiente video vamos a para abrirlo y explorar cómo es la estructura que tiene esto. Nos vemos.